BBA: An Efficient Batch Bundle Authentication Scheme for Delay Tolerant Networks

Haojin Zhu[†], Xiaodong Lin[‡], Rongxing Lu[†], Xuemin (Sherman) Shen[†] and Pin-Han Ho[†]
[†]Department of Electrical and Computer Engineering, University of Waterloo, Canada N2L 3G1
[‡]Faculty of Business and Information Technology, University of Ontario Institute of Technology, Canada L1H 7K4
Email: {h9zhu, rxlu, xshen, pinhan}@bbcr.uwaterloo.ca, xiaodong.lin@uoit.ca

Abstract— **To realize efficient in-transit messages (bundles) authentication in delay tolerant networks (DTNs), this paper introduces a novel batch bundle authentication (BBA) scheme to validate the bundles in a batch instead of authenticating them one by one. We take the advantage of identity based cryptography to dramatically reduce the transmission cost, and adopt batch signature technique to realize the efficient bundle signature verification. Compared with existing message authentication approaches, our scheme has the superiority on improved efficiency even under the invalid signature attack. Simulation results demonstrate that the proposed scheme can be an enhancement for current bundle security protocol specification.**

I. Introduction

Most of the popular network applications on the Internet today are built on the assumption of existence of a contemporaneous end-to-end link between the source and destination. However, there are many cases for which such an assumption is invalid, such as space communication and networking in sparsely populated areas [1], vehicular ad hoc networks [2], [3] and underwater networks [4]. These new emerging networks characterized by long propagation delays and/or intermittent connectivity are often referred to as Delay Tolerant Networks (DTNs). In DTNs, the in-transit messages (bundles) could be sent over an existing link, get buffered at the next hop until the next link in the path appears (e.g., a new node moves in range or an existing one wake-up), and so on and so forth, until it reaches its destination. This message propagation process is usually referred to as "store-carry-and-forward" strategy and the routing is made in "opportunistic" fashion.

Many previously reported studies have focused on opportunistic routing issues in DTNs [4]. However, the security issues, especially on how to ensure the authenticity and integrity of bundles in DTNs, have given little attention. In DTNs, malicious routers can arbitrarily insert false information into the bundles. If innocent routers further propagate these forged messages, the attacks may generate large amounts of unwanted traffic to the network. Due to resource-scarcity characteristic of DTNs, the extra traffic may pose a serious threat on the operation of DTNs [5]. Further, unauthorized access and utilization of DTN resources are another serious concern in terms of traffic authentication.

The current primary security proposal, *bundle security protocol specification* [6] proposed by delay tolerant networking research group (DTNRG), suggests to adopt Bundle Authentication Block (BAB) to provide authentication for single hop by adding a digital signature to the bundle. More specifically, both users and forwarding routers are issued public key certificates and key pairs, where the certificates indicate the class-of-service (CoS) right of users. A bundle sender can sign the bundles with its private keys and produce a bundle-specific digital signature. This signature allows receivers as well as intermediate forwarders to confirm the authenticity of senders, the integrity of the messages and the sender's CoS rights. We refer this bundle authentication method based on verifying every individual bundle signature as the *individual bundle authentication scheme*.

However, when it comes to deployment, individual bundle authentication scheme has faced two main obstacles: time and space. Firstly, the size of digital signatures and the corresponding public key certificates is typically very large in the order of tens (using Elliptic Curve Cryptography) to hundreds of bytes (RSA), which will introduce extra transmission overhead. Secondly and more importantly, public key signature verification is typically computational extensive operations, and thus verifying those individual signatures one by one at each intermediate DTN router can significantly slow down the time it takes for route to propagate the bundles. These extra communication and transmission overhead may present a great challenge to the security design in DTNs. This situation will be worsen when flooding or multi-copy based propagation method is employed to enhance the reliability of DTN transmission [4], since the signature verification operations will be performed along each data delivery path. In [8], it is suggested that a bundle sender collects all "unsigned" bundles, builds a Merkle hash tree [9] on them, and signs the root of the tree. Thus, it generates one signature for all unsigned bundles, instead of one for each bundle. This approach can reduce the computational cost at the expense of increased transmission size. However, it cannot be applied to authentication on bundles from different senders.

On the other hand, the unique "store-carry-and-forward" transmission characteristic of DTNs implies that multiple bundles from distinct/common senders may be accumulated at some common intermediate nodes. For these nodes, verifying signatures in batch instead of one by one is more suitable for bundle authentication in DTNs. Therefore, based on this fact, in this paper, we propose a novel batch bundle authentication scheme (BBA) to efficiently validate the bundles in DTNs. BBA can reduce both the transmission and computational overhead by adopting identity based signature with batch verification technique, which is first introduced in [11]. BBA enables the verification cost of signatures to be constant regardless of the size of the batch. Further, to tolerate *invalid signature injection attack*, which is defined as an attacker

978-1-4244-2324-8/08/$25.00 © 2008 IEEE.

This full text paper was peer reviewed at the direction of IEEE Communications Society subject matter experts for publication in the IEEE "GLOBECOM" 2008 proceedings.

may intentionally inject invalid signatures to the network to circumvent the batch verification technique, we adopt the "divide-and-conquer" approach to efficiently detect invalid signature. Simulation results show that BBA is more efficient than individual bundle authentication scheme even if up to 10% of the signatures are invalid.

The remainder of the paper is organized as follows. In Section II, some related work is reviewed. In Section III, we present the system model, attack model, and the design goals. In Section IV-C, the proposed BBA scheme is presented in detail. Performance analysis is given in Section V, followed by the conclusion in Section VI.

II. Related Works

It has been widely recognized that security issue is one of the major challenges for DTN design. In the current "Bundle Security Protocol Specification" [6], there are two types of security blocks which are related to bundle authentication: the Bundle Authentication Block (BAB) and the Payload Integrity Block (PIB), where BAB is used to assure the authenticity and integrity of the bundle along a single hop from forwarder to intermediate receiver while PIB is used to assure the authenticity and integrity of the bundles from the source to the destination. In other words, BAB is used to protects a bundle on a "hop-by-hop" basis and PIB protects on a "end-to-end" basis. The detailed authentication procedure can be illustrated by Fig. 1. The source S sends its bundle, together with its bundle-specific signature sig_S and the certificate $Cert_S$ to an adjacent forwarding node B. The forwarding node B verifies the sender's identity and CoS rights by verifying sender S's signature and certificate. Then, the forwarding node B replaces the sender's signature with its own signature and then forwards it to the next DTN router C. Each subsequent DTN router verifies the signature as well as certificate of the sender and previous forwarding node. Then, it replaces the prior node's signature with its own signature as well as certificate and forwards it to the next forwarding router.

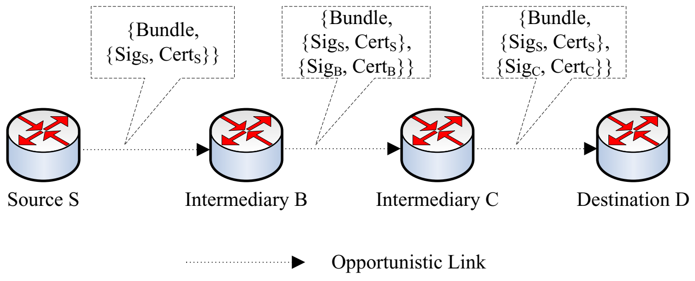

Fig. 1. Hop-by-Hop Bundle Authentication in DTN

One of the major problems of this approach is that it does not perform well in cases that fragmentation of bundles is needed because the receiver cannot authenticate any of received fragments if it has not yet received the entire message. To address this problem, one approach called as "toilet paper" was proposed in [12]. The main idea is to make each fragment self-authenticating by attaching a signature to the end of each fragment separately. However, this approach will lead to a more serious performance issue since the DTN routers have to spend more computational effort on verifying growing number of signatures. In [8], it is suggested to use binary hash tree based authentication to reduce the computational cost. However, the verifier needs to perform a few extra hashing operations when verifying a fragmentation signature, and the message size grows (due to the attached hash path). Other research effort on DTN security includes [1] and [7], which discuss the anonymity and security issues in DTNs.

III. Models and Design Goals

This section describes our system model and attack model, followed by design goals.

A. System Model

We model a DTN as a directed graph $G = (V, E)$, where V and E represent the set of nodes and edges, respectively. A source node can deliver packets to a destination node via one or multiple pathes depending on any particular forwarding algorithm such as [4]. Specifically, as shown in Fig. 2, for a given intermediate node F, it may contemporarily receive bundles $\{M_{ij} | 1 \le j \le k_i\}$ from multiple bundle senders $\{ID_i | 1 \le i \le n\}$ via one or multiple hops. The received packets $\{M_{ij} | 1 \le i \le n, 1 \le j \le k_i\}$ will be buffered until the next link in the path appears. We assume that an *Offline Security Manager (OSM)* exists in our scheme. Before joining the DTN network, every DTN node should register to the OSM and obtain its corresponding ID based secret key.

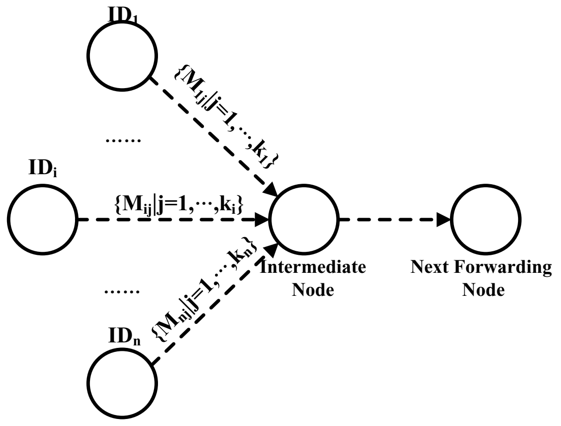

Fig. 2. Bundles buffered at intermediate node

B. Threat Model

We focus on the *false message injection attack* and *invalid signature injection attack*. In these attacks, the adversary can generate invalid messages (or false message injection attack) as well as invalid supporting signatures (or invalid signature injection attack) to DTN networks, which makes intermediate DTN nodes have to spend a lot computational efforts or transmission efforts in verifying these false messages.

C. Design Goals

The design goals include:

- *Effectiveness*: The proposed scheme should be effective in containing the damage of false message injection attack.
- *Efficiency*: The proposed bundles authentication scheme should be performed in an efficient way to reduce the communication and transmission overhead.

IV. Batch Bundle Authentication Scheme

In this section, we first provide some preliminary background, and then present in detail the BBA scheme.

A. Pairing Technique

The proposed BBA scheme is based on bilinear pairing which is briefly introduced as below. Let $\mathbb{G}$ be a cyclic additive group and $\mathbb{G}_T$ be a cyclic multiplicative group of the same order q, i.e., $|\mathbb{G}| = |\mathbb{G}_T| = q$. Let P be a generator of $\mathbb{G}$. We further assume that $\hat{e} : \mathbb{G} \times \mathbb{G} \rightarrow \mathbb{G}_T$ be an efficient admissible bilinear map with the following properties:

- Bilinear: for $a, b \in \mathbb{Z}_q^*$, $\hat{e}(aP, bP) = \hat{e}(P, P)^{ab}$.
- Non-degenerate: $\hat{e}(P, P) \neq 1_{\mathbb{G}_T}$.
- Computable: there is an efficient algorithm to compute $\hat{e}(P_1, Q_1)$ for any $P_1, Q_1 \in \mathbb{G}$.

According to [10], such an admissible bilinear map $\hat{e}$ can be constructed by Weil or Tate pairings on the elliptic curves.

B. Identity Based Signature with Batch verification

Identity-based cryptography (IBC) is a powerful alternative to traditional certificate-based cryptography. Its main idea is to make an entity's public key directly derivable from its publicly known identity information such as the e-mail address. Eliminating the need for public-key certificates and their management makes IBC much more appealing for securing DTNs, where the need to transmit and check certificates has been identified as a significant limitation.

The main computation cost for authenticating the bundles comes from verifying a set of bundle-specific signatures issued by different bundle senders. The corresponding public key certificates of the signers also need to be verified together. All of them will incur a significant amount of transmission and verification cost. In this study, we take advantage of ID-based signature to reduce the transmission cost of bundle-specific signatures, while relieves the verifier to check the authenticity of public key certificates. Further, we take advantage of batch verification technique to reduce the verification cost. A batch signature is a digital signature that enables the verifiers to quickly verify a set of digital signatures on different messages by different signers. In this study, we adopt ID-based signature with batch verification introduced in [11] (namedly CC scheme) as our basic cryptographic tool to improve the verification performance.

C. BBA Scheme

BBA scheme has four steps. In the "System Initialization" step, the OSM chooses its public/private key pairs and generates secret key for every DTN node. A bundle sender will generate bundle specific signatures through "Bundle Signature Generation" step. When multiple bundles accumulate at an intermediary DTN node, the signatures verification operation can be performed in "Batch Verification on Bundle Signature" step. Finally, "Detection of Invalid Bundle Signatures" step is developed to contain invalid signature injection attack.

1) System Initialization: The BBA scheme adopts similar bilinear pairing system parameters $(q, \mathbb{G}, \mathbb{G}_T, \hat{e}, P)$ in [11]. The OSM picks a random number $s \in Z_q^*$ as the system master key and computes $P_{pub} = sP$ as the public key. In addition, two hash functions are formed: $H_1 : \{0,1\}^* \times \mathbb{G} \rightarrow \mathbb{Z}_q^*$ and $H_2 : \{0,1\}^* \rightarrow \mathbb{G}$. The public key and system parameters (P, P_{pub}, H_1, H_2) will be preloaded in every DTN node. For any DTN node which is going to join the DTN system, it can also randomly choose $\mathcal{ID}_i$ as its pseudonym and contact the OSM to obtain its corresponding secret key. The OSM will verify its identity, determine its class-of-service (CoS) right CoS_i and then compute the secret key $sk_i = sH_2(\mathcal{ID}_i || CoS_i)$. Note that, $pk_i = H_2(\mathcal{ID}_i || CoS_i)$ serves as the public key of DTN node $\mathcal{ID}_i$.

2) Bundle Signature Generation: When a bundle sender is going to generate a bundle, it signs on the bundles with its private keys. The intermediate node on the way to the destination can verify the signature to check the authenticity and integrity of the bundles and make sure that the bundle sender is an authorized DTN node. The bundle signing algorithm is shown as follows.

1) **Sign** Given a particular DTN node $\mathcal{ID}_i$ and a bundle M_j, $\mathcal{ID}_i$ can generate a signature $Sig_{ij} = (U_{ij}, V_{ij})$ by computing: $U_{ij} \leftarrow rpk_i$, $h \leftarrow H_1(M_j, U_{ij})$ and $V_{ij} \leftarrow (r + h)sk_i$, where r is a randomly chosen number.
2) **Verify** The intermediate forwarding node can verify the signature $Sig_{ij} = (U_{ij}, V_{ij})$ by computing: $h_j \leftarrow H_1(M_j, U_{ij})$ and accept it as a valid signature if $\hat{e}(P, V) = \hat{e}(P_{pub}, U_{ij} + h_j pk_i)$.

Regarding to the transmission overhead, since identity based cryptography is adopted, the intermediate nodes can be relieved from public key certificate transferring and verification, which may greatly improve the bundle authentication performance. On the computational overhead, the computation cost for verifying a single signature is dominantly comprised of two pairing operations. Note that the computation cost of a pairing operation is much higher than the cost of an addition and a multiplication operation. In the following section, we will discuss how to take advantage of batch verification to further improve the bundle authentication efficiency.

3) Batch Verification on Bundle Signatures: In DTNs, there are three cases that multiple Bundles may accumulate at an intermediary DTN node, which will be summarized as follows:

- **Case I: Bundles from different senders** A given intermediate node F may contemporarily receive bundles send by different bundle senders. These bundles originated by different senders may be buffered at this intermediate node before the next link in the path appears.

- **Case II: Bundles from the same sender** A given intermediate node F may receive bundles from a single bundle senders. This is because fragmentation of bundles is often needed in DTN. To support bundle fragmentation, proactive fragmentation allows the bundle sender deliberately to break an entire bundle into smaller pieces and each fragment includes a signature to make it be self-authenticated by the intermediate nodes. The fragmentation makes the intermediate nodes more easily to received multiple bundles from a common sender.
- **Case III: Hybrid of cases I and II** In a realistic situation, it is more common for an intermediate node to receive bundles from distinct senders or a common sender, which may be considered as the combination of Case I and Case II.

In summary, the bundle accumulation at the intermediate nodes stems from the intrinsic "store-carry-and-forward" transmission nature of DTN networks, which allows us to take advantage of batch verification to reduce the computational cost of bundle verification.

Generally, given n distinct bundle authentication blocks $\{M_{ij}, Sig_{ij}, \mathcal{ID}_i\}$, where $i \in \{1, \ldots, n\}$ and $j \in \{1, \ldots, k_i\}$ refer to the index of DTN node and the index of the message sent by DTN $\mathcal{ID}_i$, respectively. With batch verification technique, the intermediate node can verify these signatures in batch by performing the following batch verification algorithm.

3) **Combination** Computes $V_{Batch} = \sum_{i=1}^{n} \sum_{j=1}^{k} V_{ij}$, $U_{Batch} = \sum_{i=1}^{n} \sum_{j=1}^{k} U_{ij} + h_j pk_i$. V_{Batch} and U_{Batch} is the combined signature.

4) **Batch Verification** Given the combined signature V_{Batch} and U_{Batch}, the bundle set $\{M_{ij} | 1 \leq i \leq n, 1 \leq j \leq k\}$ on which it is based for all senders $\{\mathcal{ID}_i | 1 \leq i \leq n\}$, the verifier can authenticate the bundles in two steps:
   - Ensure that the bundle senders have the appropriate CoS rights.
   - Accept if $\hat{e}(P, V_{Batch}) = \hat{e}(P_{pub}, U_{Batch})$ holds.

For more detailed discussion on security analysis of this batch signature, see [11]. It is observed that the computation cost that the intermediate node spends on verifying n signatures is dominantly two paring operations. This appealing property demonstrates that the verification time for multiple signatures is constant regardless of the size of the batch. Thus, this batch verification can dramatically reduce the verification delay, particularly when verifying a large number of signatures.

4) Detection of Invalid Bundle Signatures: Batch verification scheme may be vulnerable to invalid signature injection attack, which is defined as a variant of false message attack in that a malicious DTN node may arbitrarily inject forged bundles and invalid bundle-specific signatures into the legitimate bundles. Under invalid signature injection attack, if there is even a single invalid signature in the batch, then the batch verifier will be rejected with high probability. Therefore, it is critical to tolerate some percentage of invalid signatures without losing the performance advantage of batch verification.

In this paper, we adopt the recursive "divide-and-conquer" approach, which is firstly investigated in [13], to address the invalid signature injection attack. The basic idea of divide-and-conquer is that the set of signatures in a failed batch is repeatedly split into several smaller sub-batches to verify. A simple version of "divide-and-conquer" approach is simple binary search, which can be summarized by Algorithms 1. We assume the $|N|$ is the size of the batch verifying signature set $\{Sig_i | 1 \leq i \leq N\}$. Let $BatchVerify(x, y)$ denote the batch verification operation of signature set $\{Sig_j | x \leq j \leq y\}$, where x and y refer to the index of lower bound and upper bound, respectively.

```
Data: Input $\{Sig_i | X \leq i \leq Y\}$
Result: Output Invalid signature
ISD($X, Y$) :
begin
    if BatchVerify(X,Y) then
        return True;
    else if X==Y then
        return $Sig_X$ as an invalid signature ;
    else
        ISD($X, \lceil (X+Y)/2 \rceil$) ;
        ISD($\lfloor (X+Y)/2 \rfloor, Y$) ;
    end
end
```

Algorithm 1: Invalid Signature Detection

The detailed performance evaluation on batch bundle verification will presented in section V-B.

V. Performance Evaluation

In this section, we evaluate the performance of the proposed BBA scheme in terms of the resultant communication cost and computation overhead. To demonstrate the superiority of BBA, we also compare BBA with two individual bundle authentication (or called as IBA) variations called ECDSA-IBA and CC-IBA, which are based on ECDSA, Cha and Cheon signature scheme [11], respectively.

A. Transmission Cost

Compared with the ECDSA signature scheme of IEEE1069.2, the length of a signature in the BBA scheme is the same as that of the ECDSA, i.e., 320 bits=40 bytes. However, due to adoption of identity based signature, BBA allows public keys to be derived from entities' known identity information, thus eliminating the need for public key distribution and certificates. In contrast, the ECDSA scheme has to incorporate a certificate in the message, which is 125 bytes long in the case of using the WAVE certificate [14].

B. Computational Cost

The computation costs are measured by the most expensive pairing (Pair) and point multiplication (Pmul) operation. Because of the batch verification technique, the total verification cost on N distinct signatures is about only 2 pairings, which is a significant improvement over the $2N$ pairings required by individual verification. This appealing property demonstrates that the verification time for multiple signatures is constant regardless of the size of the batch. By adopting the pairing technique in [15], we can obtain the rough time costs for Pair and Pmul to be 2.82 ms and 0.78 ms, respectively. In Table I, we compare the transmission cost and computational cost of various schemes. Here, N refers to total size of batch.

TABLE I

COMPARISON OF TRANSMISSION AND COMPUTATION COST

	ECDSA-IBA	CC-IBA	BBA
Transmission Cost (bytes)	$(40 + 125)N$	$40N$	$40N$
Computation Cost (ms)	$0.78 * 4N$	$2.82 * 2N$	$2.82 * 2$

C. Impact of Invalid Signature Injection Attack on Computation Cost of BBA

We also evaluate the computational cost of BBA under the invalid signature injection attack. If there is only one invalid signature in the batch, the best cost to detect a single invalid signature is $2\lceil logN \rceil$. In general, given w invalid signatures and the total batch size of N, according to [16], the worst cost of BBA scheme can be summarized as follows:

$$Cost(w, N) = \begin{cases} 4 * Cost_{\text{Pairing}}(2^{\lceil \log_2 w \rceil} - 1 \\ \quad +w * (\lceil \log_2 N \rceil - \lceil \log_2 w \rceil)), & \text{if } 1 \leq w \leq \dfrac{N}{2} \\ \dfrac{N}{2} * Cost_{\text{Pairing}}, & \text{if } \dfrac{N}{2} \leq w \leq N \end{cases}$$

where $Cost_{\text{Pairing}}$ refers to the cost of performing a pairing operation. As a comparison, since the signatures are verified one by one, ECDSA-IBA and CC-IBA do not need to perform extra operations to detect invalid signatures. Therefore, in Fig. 3, we compare BBA scheme with ECDSA-IBA and CC-IBA scheme under the presence of different percentages of invalid signatures. It can be seen that, when the percentage of invalid signatures is small, the proposed BBA yields the shortest computation latency, while the worst verification cost will grow dramatically along with the increasing percentage of invalid signatures. This result also demonstrates the recent empirical study conducted in [17], which indicates that if less than 10% of the signatures are invalid, then batch verification is still more efficient than individual verification.

VI. CONCLUSION AND FUTURE WORK

In this paper, we have proposed an efficient bundle authentication scheme for delay tolerant networks, which can effectively reduce the transmission cost as well as computational cost. Our future research will consider how to implement the proposed scheme in a realistic delay tolerant environment and how to optimize the batch verification cost by adopting different message transmission strategy.

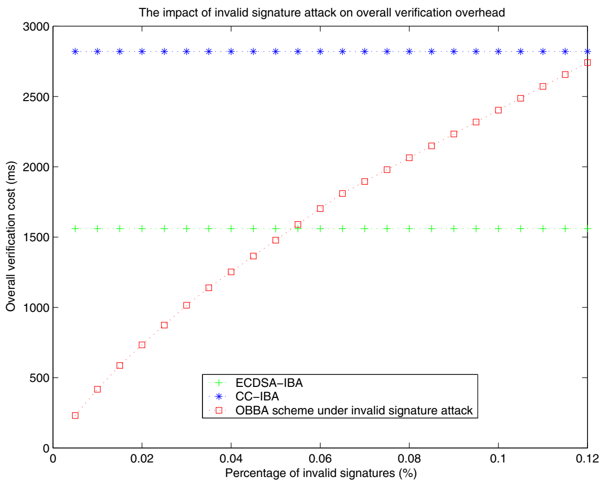

Fig. 3. Impact of invalid signature attack on verification cost

REFERENCES

[1] A. Kate, G. Zaverucha and Urs Hengartner, "Anonymity and security in delay tolerant networks," Proc. of *SecureComm 2007*, Sept. 2007.

[2] H. Zhu, X. Lin, R. Lu, P.H. Ho, and X. Shen, "AEMA: An aggregated emergency message authentication scheme for enhancing the security of vehicular ad hoc networks," Proc. of *IEEE ICC'08*, Beijing, China, May 19-23, 2008.

[3] C. Zhang, R. Lu, X. Lin, P.H. Ho and X. Shen, "An Efficient Identity-based Batch Verification Scheme for Vehicular Sensor Networks," Proc. *IEEE INFOCOM'08*, Phoenix, AZ, USA, April 14-18, 2008.

[4] T. Spyropoulos, K. Psounis and C. S. Raghavendra, "Efficient routing in intermittently connected mobile networks: the multiple-copy cast," *IEEE/ACM Trans. on Networking*, vol. 16, no. 1, Feb. 2008.

[5] S. Farrell and V. Cahill, "Security consideartons in space and delay tolerant networks," Proc. of *SMC-IT'06*, July 2006.

[6] S. Symington, S. Farrell, H. Weiss and P. Lovell, "Bundle security protocol specification," draft-irtf-dtnrg-bundle-security-05.txt, work-in-progress, February 2008.

[7] M. Shi, K. AlMotairi, X. Shen, J.W. Mark, and D. Zhao, "Credit-Based User Authentication for Delay Tolerant Mobile Wireless Networks," Proc. of *IEEE ICC'08*, Beijing, China, May 19-23, 2008.

[8] N. Asokan, K. Kostiainen, P. Ginzboorg, J. Ott and Cheng Luo, "Applicability of identity-based cryptography for disruption-tolerant networking," Proc. of *the First International MobiSys Workshop on Mobile Opportunistic Networking (MobiOpp)*, June 2007.

[9] R. Merkle, "Protocols for public key cryptosystems," Proc. of *IEEE S&P*, pp. 122-133, 1980.

[10] D. Boneh and M. Franklin, "Identity Based Encryption from the Weil Pairing," Proc. of Crypto'01, LNCS, vol. 2139, pp. 213-229, Springer-Verlag, 2001.

[11] J. C. Cha and J. H. Cheon, "An identity-based signature from gap Diffie-Hellman groups," Proc. of *PKC'03*, vol. 2567, pp. 18-30. 2003.

[12] DTNRG. Delay tolerant networking research group: dtn-interest mailing list archive, April 2005. Available from http://mailman.dtnrg.org/pipermail/dtn-interest/2005-April/.

[13] J. Pastuszak, D. Michatek, J. Pieprzyk, and J. Seberry, "Identification of bad signatures in batches," Proc. of *PKC'00*, vol. 3958, pp. 28-45, Springer-Verlag, 2000.

[14] IEEE Std 1609.2-2006. IEEE Trial-Use Standard for Wireless Access in Vehicular Environments - Security Services for Applications and Management Messages, 2006.

[15] P. Barreto, B. Libert, N. McCullagh and J.-J. Quisquater, "Efficient and provably-Secure identity-based signatures and signcryption from bilinear maps," Proc. of *AISACRYPT 2005*, pp. 515-532, 2005.

[16] L. Law and B. J. Matt, "Finding invalid signatures in pairing-based batches," Proc. of *IMA Int. Conf.*, vol. 4887, LNCS, pp. 34-53, 2007.

[17] A. L. Ferrara, M. Green, S. Huhenberger and M. Pedersen, "On the practicality of short signature batch verification," available in http://eprint.iacr.org/2008/015.pdf, 2008.